

New Tools For Your Threat Hunting Toolbox

Mark Baggett

@MarkBaggett

Visit sans.org/free for thousands of FREE Cybersecurity Resources

```
Get-ADUser -Filter "Mark Baggett" | fl -Properties *
```

- Mark Baggett
- Penetration Testing and Incident Response Consulting
- Senior SANS Instructor
- Author of SANS SEC573 Automating InfoSec with Python
- Masters in Information Security Engineering
- GSE #15
- DoD Advisor, Former CISO 18+ years commercial

```
student@573:/opt/metasploit-framework$ grep -Ri "mark baggett" | wc -l  
7
```

The Intrusion Simulation Lab

- Malware behavior simulator
- Train your incident responders to find malware in your environment based on common malware behavior
- Launches a known behavior and then ask you to identify it
- Randomized port numbers and processes each time you run it
- Teaches the technique used on the SEC504 incident handling cheat sheets which are available for free!

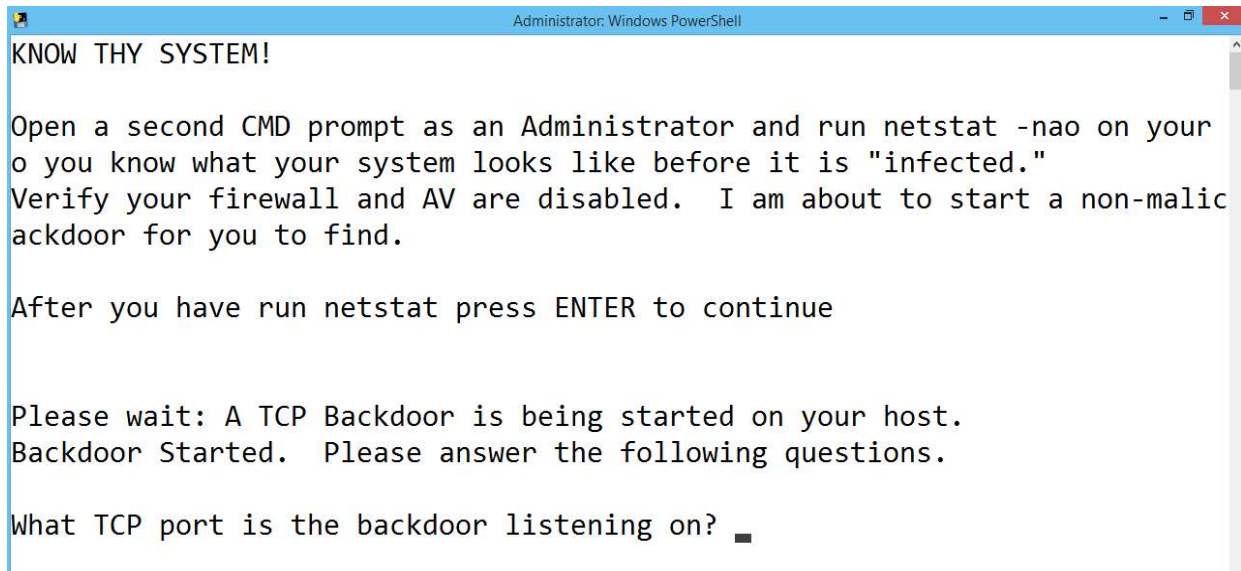
**SEC504: Hacker Tools,
Techniques, Exploits, and
Incident Handling**



GIAC Certified Incident Handler (GCIH)

Several Techniques for Your Team to Uncover

- Listening Backdoors
- HTTP Backdoor
- PowerShell Backdoor
- Command inspection
- Base64 code obfuscation



```
Administrator: Windows PowerShell

KNOW THY SYSTEM!

Open a second CMD prompt as an Administrator and run netstat -nao on your
o you know what your system looks like before it is "infected."
Verify your firewall and AV are disabled. I am about to start a non-malic
ackdoor for you to find.

After you have run netstat press ENTER to continue

Please wait: A TCP Backdoor is being started on your host.
Backdoor Started. Please answer the following questions.

What TCP port is the backdoor listening on? █
```

- Download the tool here: <https://markbaggett.github.io/504lab/>

Packet Fragmentation Reassembly Engines with REASSEMBLER

- Reassembler gives you SOC analysts insight into what your NDR/IPS/IDS sees
- Today most IDS's when setup properly faithfully reproduce the various OS packet reassembly engines to find attacks!
- As of August 2018 Linux rejects overlapping fragments, but Windows and MacOS still enable these attacks
- Reassembler give analysts the ground truth of what happened on the network

SEC503: Intrusion Detection In-Depth



GIAC Certified Intrusion Analyst (GCIA)

One Packet In -> Six Packets Out!

```
(573) student@SEC573:~/Desktop/reassembler$ python reassembler.py final_frags.pcap
```

```
Reading fragmented packets from disk.
```

```
Packet fragments found. Collecting fragments now.
```

```
Reassemble packets between hosts 172.16.120.191 and 172.16.120.1? [Y/n] y
```

```
Reassembled using policy: First (Windows, SUN, MacOS, HPUX)
```

```
GET /etc/passwd
```

```
Host:www.supersecret.net
```

```
User-Agent:evil-browser
```

What Windows saw

FRAGMENTS IN!

```
Reassembled using policy: Last/RFC791 (Cisco)
```

```
GET /not/catdog
```

```
Host:www.supersegway.org
```

```
User-Agent:good-browser
```

What Cisco sees

```
Reassembled using policy: Linux (Umm.. Linux)
```

```
GET /etc/catdog
```

```
Host:www.supersecret.net
```

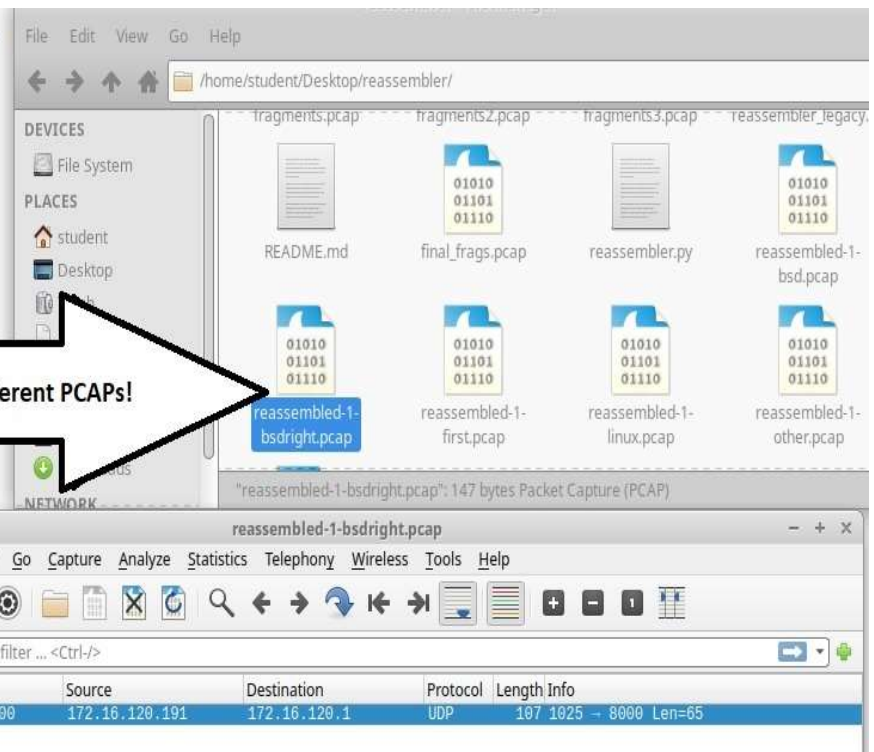
```
User-Agent:good-browser
```

What Linux saw 2018

```
Reassembled using policy: BSD (AIX, FreeBSD, HPUX, VMS)
```

```
GET /etc/passwd
```

Writes 6 different PCAPs!



- Download from <http://github.com/markbaggett/reassembler>

Identify DGA Names Used by attackers with Freq.py

- DGA names are used in
 - Command and Control Channels
 - Malicious Payloads by droppers
 - Certificate names in malware
- Accurately random looking names such as those from this recent headlining breach

SEC511: Continuous Monitoring and Security Operations



GIAC Continuous Monitoring Certification (GMON)

- External forwarding address:

```
daemon [@ daemongr5yenh53ci0w6cjbbh1gy1161fxpd.com
```

- Link contained in the malicious 0365 add-in:

```
https: // iwljzmwhres67fh[.]com / office
```


Frequency Score of Legitimate Domains

- Uses two methods to score how "Normal" a string is
- Normal strings score Method 1 > 5 and Method 2 > 4

```
student@573:~/Desktop/freq$ python freq.py -m google.com freqtable2018.freq
(6.6009, 4.9975)
student@573:~/Desktop/freq$ python freq.py -m youtube.com freqtable2018.freq
(10.3381, 6.881)
student@573:~/Desktop/freq$ python freq.py -m reddit.com freqtable2018.freq
(8.8356, 8.5714)
student@573:~/Desktop/freq$ python freq.py -m slack.com freqtable2018.freq
(5.7657, 5.189)
student@573:~/Desktop/freq$ python freq.py -m instagram.com freqtable2018.freq
(7.5582, 7.3355)
```


Frequency of Suspicious Domains

- Scores for URLs used in know malware are much lower!
 - Method 1 < 5
 - Method2 < 4



File Edit View Terminal Tabs Help

```
student@573:~/Desktop/freq$ python freq.py -m ukvkloytfaw.bid freqtable2018.freq
(2.2847, 2.1507)
student@573:~/Desktop/freq$ python freq.py -m xcukrfpchsxn.com freqtable2018.freq
(4.1311, 3.2014)
student@573:~/Desktop/freq$ python freq.py -m ybrjldiexlqb.com freqtable2018.freq
(3.3749, 3.589)
student@573:~/Desktop/freq$ python freq.py -m bbqqjejhhd.bid freqtable2018.freq
(3.3332, 1.5073)
student@573:~/Desktop/freq$ python freq.py -m xct31.net freqtable2018.freq
(4.8265, 3.3812)
```

Freq_server makes Freq.py Available via JSON to your SEIM

- Freq_server makes freq scores available in a high performance web platform with JSON responses
- Used today by popular Network Monitoring Systems!

Security Onion
SOFELk
ZEEK

<http://freqserver/x123123.com>

(3.12, 4.12)



- Download from <http://github.com/markbaggett/freq>

Finding Baby Domains with DOMAIN_STATS!

- Malware domain are typically much "younger" than legitimate domains!
- Looking up every domain via whois will get you blocked
- Querying whois from a SEIM is a non-trivial problem to solve because of speed and high frequency of host names appearing in data
- Domain_stats.py attempt to solve these problems and enrich SEIM data with monitored whois data
 - Avoids whois queries by localizing data, providing a high speed cache, using RDAP and SANS ISC as a proxy for whois data.
 - Provides an easy to use API for SEIM integration

SEC555: SIEM with Tactical Analytics



GIAC Certified Detection Analyst (GCDA)

"Normal" Domain Creation Dates

```
Terminal - student@573: ~  
File Edit View Terminal Tabs Help  
student@573:~$ whois google.com | grep "Creation"  
    Creation Date: 1997-09-15T04:00:00Z  
student@573:~$ whois youtube.com | grep "Creation"  
    Creation Date: 2005-02-15T05:13:12Z  
student@573:~$ whois reddit.com | grep "Creation"  
    Creation Date: 2005-04-29T17:59:19Z  
student@573:~$ whois slack.com | grep "Creation"  
    Creation Date: 1992-10-21T04:00:00Z  
student@573:~$ whois snapchat.com | grep "Creation"  
    Creation Date: 2012-02-28T19:29:26Z
```

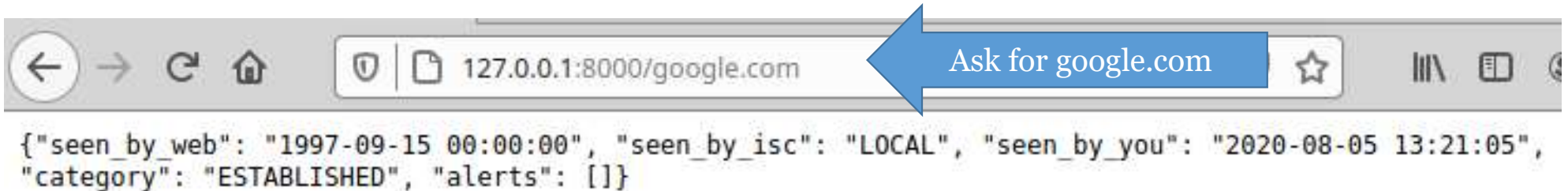
"Baby Domain" Registrations

- Of the three URLs used in previously mentioned headlining attack two of the URLs were baby domains!

```
root@ubuntu:~# whois hostwindsdns.com | grep -i "creation"
  Creation Date: 2011-05-12T23:01:53Z
Creation Date: 2011-05-12T23:01:53.00Z
root@ubuntu:~# whois iwljzmvhres67fh.com | grep -i "creation"
  Creation Date: 2020-06-24T17:17:54Z
Creation Date: 2020-06-24T17:17:54.00Z
root@ubuntu:~# whois daemongr5yenh53ci0w6cjbh1gy1l61fxpd.com | grep -i "creation"
  Creation Date: 2020-07-24T08:27:01Z
Creation Date: 2020-07-24T08:27:01Z
```

- Attackers used these domains on 7-24-2020 in a targeted attack only A FEW HOURS after the registration

Domain_stats In Action - Normal Domains



- "seen_by_web" is domain registration date
- "seen_by_isc" - Local, RDAP or date the ISC first saw the domain
- "seen_by_you" is the date your organization first saw this domain used
- Category:
 - ESTABLISHED - means registration is > 2 years old
 - NEW - means it is a newly registered domain and deserves some scrutiny
- Alerts: "Your First Contact", "ISC First Contact" and more

Domain_stats In Action - New to you

- Here is the first time we ever lookup `runcode.ninja`

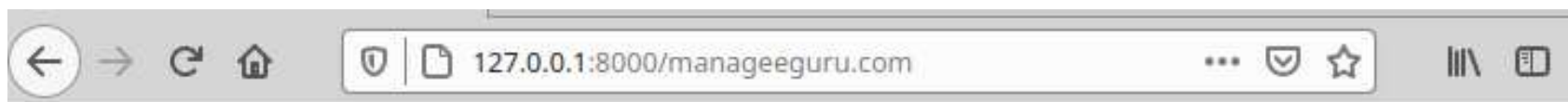


- This is the second



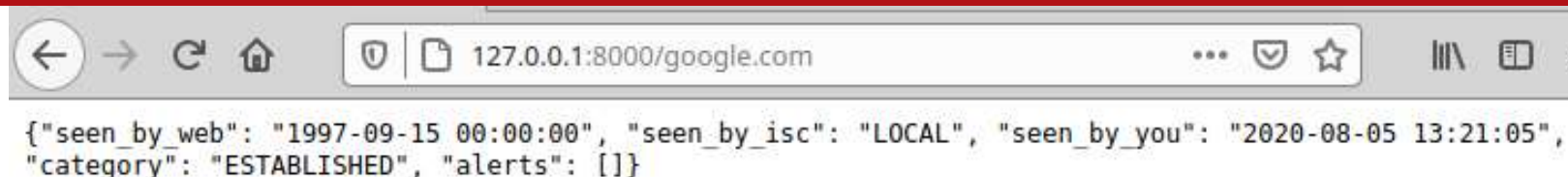
New to you, New to the world

- Here are the result from a few domain identified as evil by malcode.com



- The "NEW" category and "YOUR-FIRST-CONTACT" alert makes these stand out from the other domains in my logs

New to you, New to the world, New to the Internet Storm Center



- This tool is ready for you to use TODAY
- Name resolution is limited to
 - LOCAL - Your localized database prepopulate with 1000s of domains.
 - RDAP - Today the protocol has limited eTLD support
- Pending Enhancement:
 - Your lookups CAN be proxied through Internet Storm center to support all domains via whois
 - This enabled community base "Seen by ISC" alerts and first seen dates

Where do you get it?

- Download from http://github.com/markbaggett/domain_stats
- Has ZEEK integration script
- Deployable as a Docker!!

```
$ docker build --tag domain_stats_image http://github.com/markbaggett/domain_stats.git
$ mkdir ~/dstat_data
$ docker run -it --rm -v ~/dstat_data:/host_mounted_dir -p 8000:10000 domain_stats_image
```

```
$ docker run -it --rm -v ~/dstat_data:/host_mounted_dir -p 8000:10000 domain_stats_
No configuration file found.
WARNING: Database not found. domain_stats.db
Database is out of date. Forcing update from 1.0 to 1.3.
|XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX| 100.00% FINISHED
|XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX| 100.00% FINISHED
|XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX| 100.00% FINISHED
Folder Initialization Complete.
Using config /host_mounted_dir/domain_stats.yaml
Using database /host_mounted_dir/domain_stats.db
Using cache /host_mounted_dir/domain_stats.cache
Server is Ready. http://0.0.0.0:8000/domain.tld
^CWeb API Disabled... <<<< HIT CONTROL-C
Control-C hit: Exiting server. Please wait..
Committing Cache to disk...
Bye!
```

Kill it when its done,
then run it in the
background

You specify folder
outside docker where
data is stored long term

APIfy - The "first look" tool

- I just want to save you from typing one command a day!
- We all have our "go to" tool that we use as a first look at a suspicious host or process
- If you don't have a "go to" command check out FOR572!
- APIfy runs that command for you, returns the results as a JSON response for you to collect in your SEIM

FOR572: Advanced Network Forensics: Threat Hunting, Analysis, and Incident Response



GIAC Network Forensic Analyst (GNFA)

Configuration is Simple!

- apiify.yaml configurations control which command is run

```
! apiify.yaml ●
! apiify.yaml
27 | #
28 | #Here is an example of a whois command
29 | base_command: whois *WEBINFO*
30 | #result_regex: Creation Date.\s+(?P<creationdate>[\d:T -]+)
31 | result_regex: (?<Creation Date.|created.)\s+(?P<creationdate>[\d:T -]+)
32 | #
```

WEBINFO taken from web URL

Optional REGEX

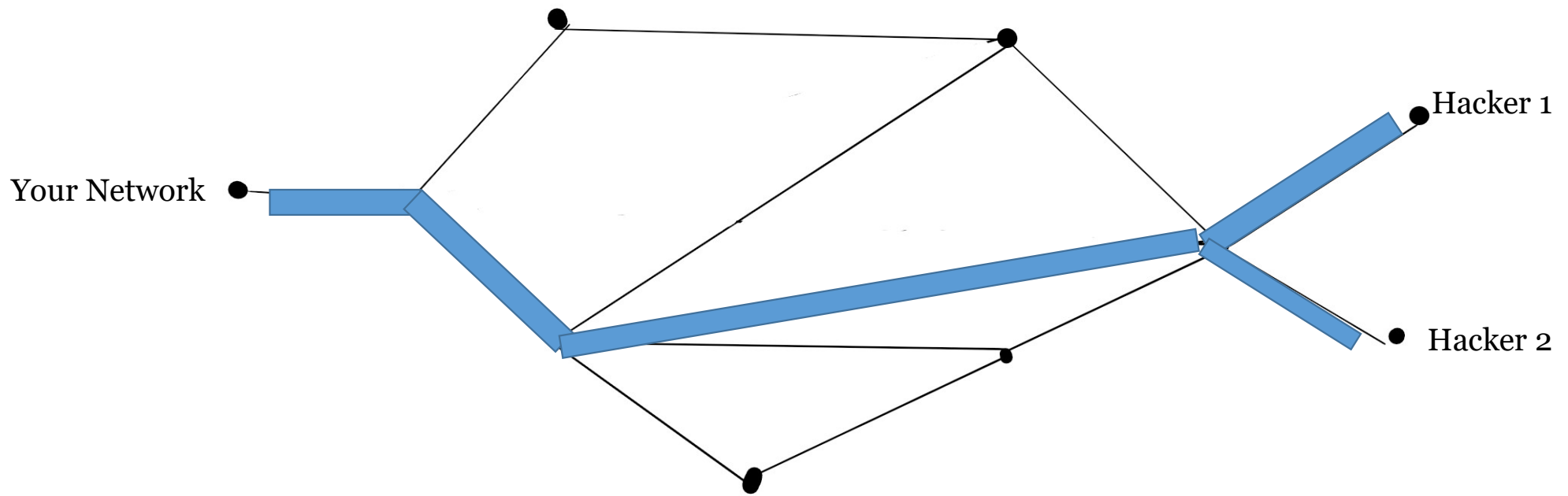
- And we are up and running

```
127.0.0.1:8000/sans.org X 127.0.0.1:8000/google.com X
127.0.0.1:8000/sans.org
Visual Python Tutor SEC573 Code Samples GeoFind Lab (Section
{"creationdate": "1995-08-04T04:00:00"}
```

WEBINFO = sans.org

'whois sans.org' regex results

Another Use Case: Traceroutes to identify shared infrastructure



traceroute #1	Hop 1 = 1.1.1.1, Hop 2 = 5.5.5.5, Hop 3 = 200.200.200.200
traceroute #2	Hop 1 = 1.1.1.1, Hop 2 = 5.5.5.5, Hop 3 = 200.200.200.200

Finding Shared Infrastructure

- I want to TRACEROUTE to every IP that generates some ZEEK alert
- Collect the path from the traceroute and record it in my SEIM
- A traceroute to google.com takes about 1 minute and 6 seconds

```
root@573:~# time traceroute www.google.com
traceroute to www.google.com (108.177.122.103), 30 hops max, 60 byte packets
 1 homefirewall.localdomain (x.x.x.1) 12.506 ms 22.639 ms 22.568 ms
...
22 108.177.122.103 (108.177.122.103) 16.196 ms * 28.300 ms

real    1m6.117s
user    0m0.000s
sys     0m0.028s
```

Make traceroute Faster!

- Traceroute has some useful options

-n	Do not resolve DNS Names for hops
-f #	Skull the first # number of hops on my side
-q #	Repeat the trace # number of times (default is 3)

- Additionally use TCP port 80 for reliability

```
root@573:~# time traceroute --tcp -p 80 -n -q1 -f3 google.com
traceroute to google.com (64.233.177.139), 30 hops max, 60 byte packets
 3 208.188.184.1 28.377 ms
21 64.233.177.139 33.881 ms
```

```
real    0m0.398s
user    0m0.000s
sys     0m0.006s
```



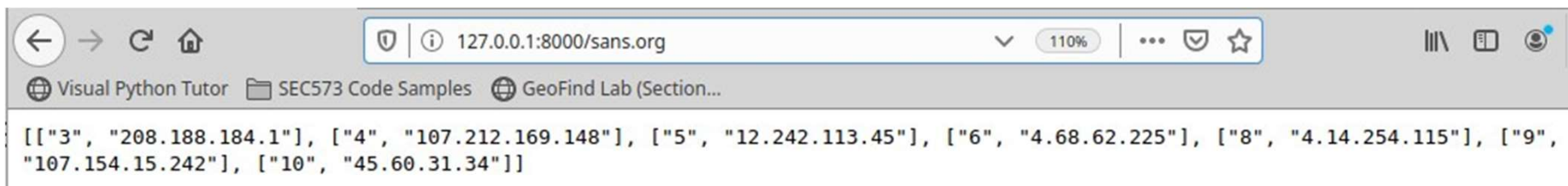
EXCEPTIONAL expect 10 seconds per
IP the FIRST TIME ONLY

Just change the apiify.yaml file!

- Another few changes to apiify.yaml

```
! apiify.yaml ●
! apiify.yaml
46 #By default the period wildcard does not match newlines. DO
47 #To use this you must uncomment all of the next 6 lines
48 base_command: traceroute --tcp -p 80 -n -ql -f3 *WEBINFO*
49 result_regex: (\d+)\s+([\d\.]+).*?$
50 regex_findall: True
51 regex_multiline: True
52 regex_ignorecase: True
53 regex_dotall: False
54
```

- And now your SIEM or ZEEK contain things like this ...



The screenshot shows a web browser window with the address bar displaying "127.0.0.1:8000/sans.org". The browser tabs include "Visual Python Tutor", "SEC573 Code Samples", and "GeoFind Lab (Section...)". The main content area displays a list of IP addresses and domain names in a JSON-like format: `[["3", "208.188.184.1"], ["4", "107.212.169.148"], ["5", "12.242.113.45"], ["6", "4.68.62.225"], ["8", "4.14.254.115"], ["9", "107.154.15.242"], ["10", "45.60.31.34"]]`.

Inspecting cached results reveals "related" IP Addresses

- Consider these three networks: 93.174.93.0, 80.82.70.0, 89.248.174.0 used by known bad actors
- Would you have suspected they share the same infrastructure?

```
student@573:~/apiify$ python dump_cache.py -s data | grep 80.82.70.0 -C1
93.174.93.0, 2020-07-29 12:36:50.062515, 9, b'["3", "208.188.184.1"], ["5",
"12.242.113.6"], ["6", "216.66.24.133"], ["7", "184.105.80.161"], ["8",
"184.105.223.166"], ["9", "72.52.92.165"], ["10", "72.52.92.214"]]'
80.82.70.0, 2020-07-29 12:37:10.092859, 50, b'["3", "208.188.184.1"], ["5",
"12.242.113.6"], ["6", "216.66.24.133"], ["7", "184.105.80.161"], ["8",
"184.105.223.166"], ["9", "72.52.92.165"], ["10", "72.52.92.214"]]'
89.248.174.0, 2020-07-29 12:37:41.493890, 1, b'["3", "208.188.184.1"], ["5",
"12.242.113.6"], ["6", "216.66.24.133"], ["7", "184.105.80.161"], ["8",
"184.105.223.166"], ["9", "72.52.92.165"], ["10", "72.52.92.214"]]'
```

APIIFY Sample Configurations include

- PING
- Entire WHOIS record
- Select Just the Creation Date from WHOIS
- Traceroute with just the last hop
- Full Traceroute to host
- Geolocation Lookup IP with Web API
- Query ISC API for IP based Threat intelligence
- Download from <http://github.com/markbaggett/apiify>

SRUM-DUMP and ESE2CSV

- Windows maintains a 30 day rolling log of all the process that was run on your end points
- This is INCREDIBLY USEFUL when you discover after the incident has occurred that your logging is otherwise insufficient.
- Even wonder "What was that thing that just popped up?" and wish something logged it?

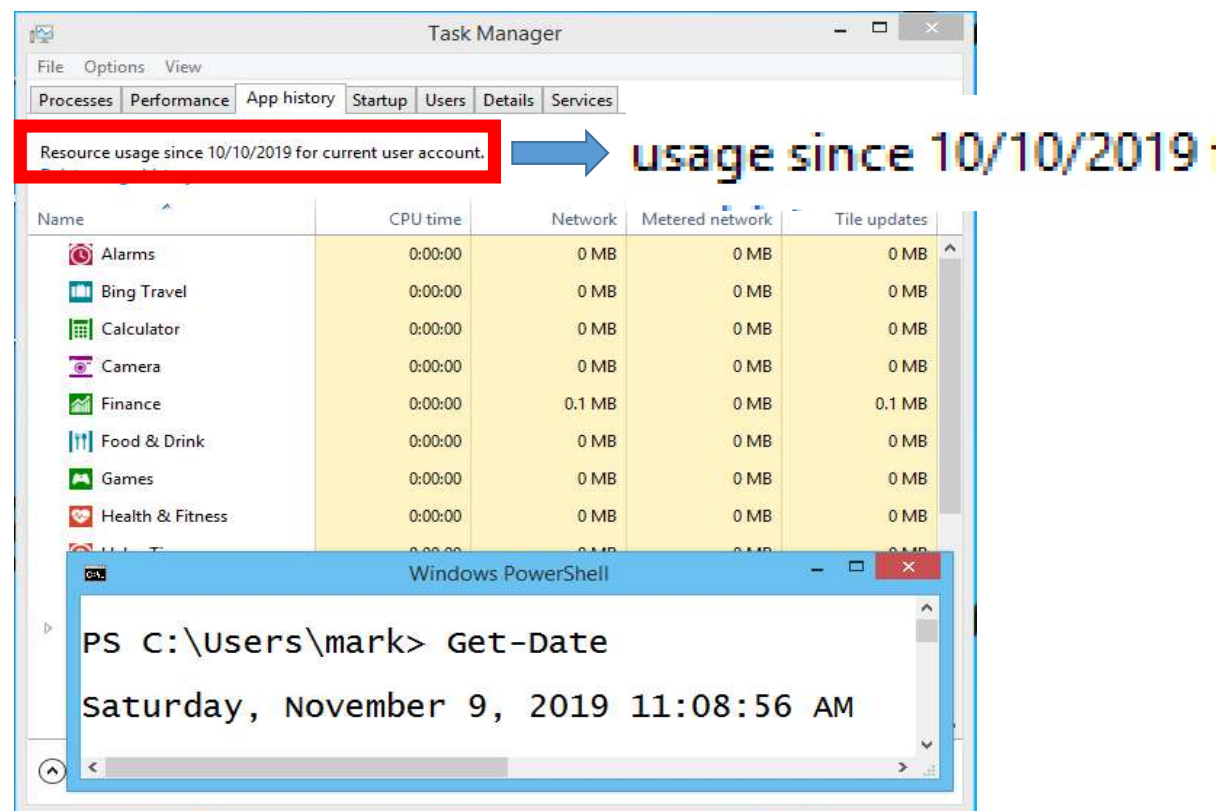
FOR500: Windows Forensic Analysis



GIAC Certified Forensic Examiner (GCFE)

Provides Access to the "APP HISTORY" tab

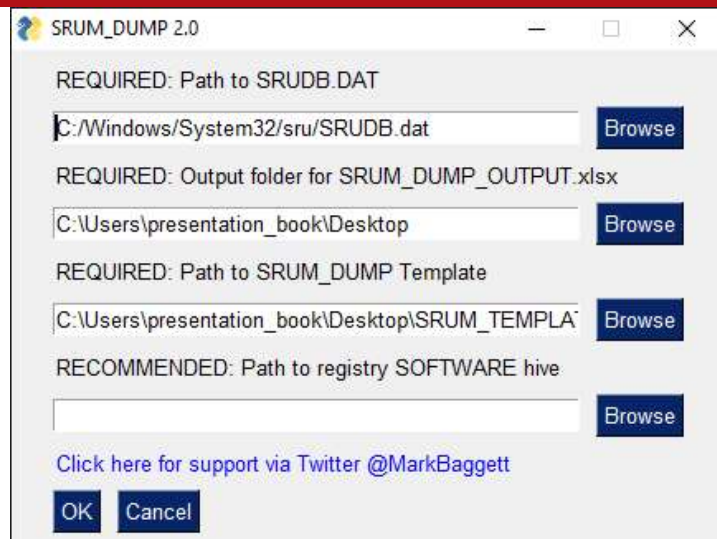
- Have you notice the "APP HISTORY" tab?
- It includes detailed logs on networks used amount of data transfer
- Stored in a database called the "SRUM"



Accessing the data in the SRUM

- SRUM-DUMP
 - Ease of Use
 - Graphical Users Interface
 - Results in Microsoft Excel
 - Only Dumps the srum database
 - <https://github.com/MarkBaggett/srum-dump>
- ESE2CSV
 - Command Line Interface for MASS collection
 - Creates CSVs for easy injection into other system
 - Dumps any database in ESE Format (srums, Edge browser history, etc)
 - Extendible and customizable via "plugins"
 - <https://github.com/MarkBaggett/ese-analyst>

SRUM-DUMP.EXE



It appears your trying to open SRUDB.DAT from a live system.
Copying or reading that file while it is locked is unlikely to succeed.
First, use a tool such as FGET that can copy files that are in use.
Try: `fget -extract c:\windows\system32\sru\sru.db.dat <a destination path>`

[Close](#) [Download FGET](#) [Auto Extract](#)

- When run with ADMINISTRATOR access it can retrieve files normally locked by the OS and analyze them.
- If you provide it a copy of the SOFTWARE registry hive it resolves usernames and network profile names for you
- Has a customizable XLSX Template that lets you customize how data is interpreted and the output format

SRUM DUMP Example Output File

- All this information is easily retrieved with SRUM-DUMP!

1	RY NUMBER	SRUM ENTRY CREATION	Application	User SID	Interface	Profile	Profile Flags	Bytes Sent	tes Received	Total Bytes
10	49979	2016-06-10 17:37:00	wlidsvc	S-1-5-21-5295836553-5295836553-529583	IF_TYPE_IEEE8021	OPENWIFI	0	6840	19113	25953
11	49980	2016-06-10 17:37:00	wlidsvc	S-1-5-19 (NT Authority)	IF_TYPE_IEEE8021	OPENWIFI	0	6968	20317	27285
12	49981	2016-06-10 17:37:00	\device\harddiskvolume\nc.exe	S-1-5-21-5295836553-5295836553-529583	IF_TYPE_IEEE8021	OPENWIFI	0	2987623984	2310	2987626294
13	49982	2016-06-10 17:37:00	None	None	IF_TYPE_IEEE8021	OPENWIFI	0	2987623984	95028474	120232533
14	49983	2016-06-10 17:37:00	CryptSvc	S-1-5-21-5295836553-529583	IF_TYPE_IEEE8021	OPENWIFI	0	8779		9335
15	49984	2016-06-10 17:37:00	Microsoft.Wi	11602.1.26.0_x64_8wekyb3S-1-5-21-5295836553-529583	IF_TYPE_IEEE8021	OPENWIFI	0	28057		33654
16	49985	2016-06-10 17:37:00								486
17	49986	2016-06-10 17:37:00								249
18	49987	2016-06-10 17:37:00								335
19	49988	2016-06-10 17:37:00								770
20	49989	2016-06-10 17:37:00								261

On the "Network Usage" tab we can easily see that "nc.exe" was used to transfer 2987623984 bytes of data over the network "OPENWIFI". Then we lookup that user's SID to see who moved data using netcat!

Network Usage Application Resource Usage Network Connections Push Notification Data Energy Usage (Long Term ...)

READY AVERAGE: 995890850.1 COUNT: 10 SUM: 5975345101

- Usually find data for EVERY process that has been run over the last 30 days!

ESE2CSV.EXE

- Processes any ese file (not just SRUM)
- Recursively search the drive for ESE files
- Acquires copies of files that are locked by the OS with -a
 - requires ADMINISTRATOR access
- List Tables in an ESE database (-l)
- Dump all tables by default (-d <optional list of tables>)
- Has plugin architecture to allow you to specify how to interpret and process ESE files
- Comes with completed SRUM template and example SPARTAN (Edge History File) template

ESE2CSV.EXE + PSEXEC.EXE = Enterprise Threat Hunting

- Use a tool such as Kape by SANS Instructor Eric Zimmerman or PSEXEC to run ESE2CSV on every host on the network
- Have ESE2CSV write output to a central network share
- Ingest all the .CSV files into a single log analysis tool to see every process that ran the environment across all machines in sequential order

```
C:\> psexec -c -u domainadmin -p pass "exe2csv.exe -a -p srudb_plugin -o \\server\share\c:\windows\system32\sru\sru.db.dat"
```

Werejugo Laptop Geolocation Tracker

- Your devices remember which wireless networks you connect to
- Windows also records the names of wireless networks that were used by processes in many different very frequently
- Various techniques can be used to determine the physical location of those wireless networks
- Putting this together we can place your laptop "at the scene of the crime" at a given date and time.

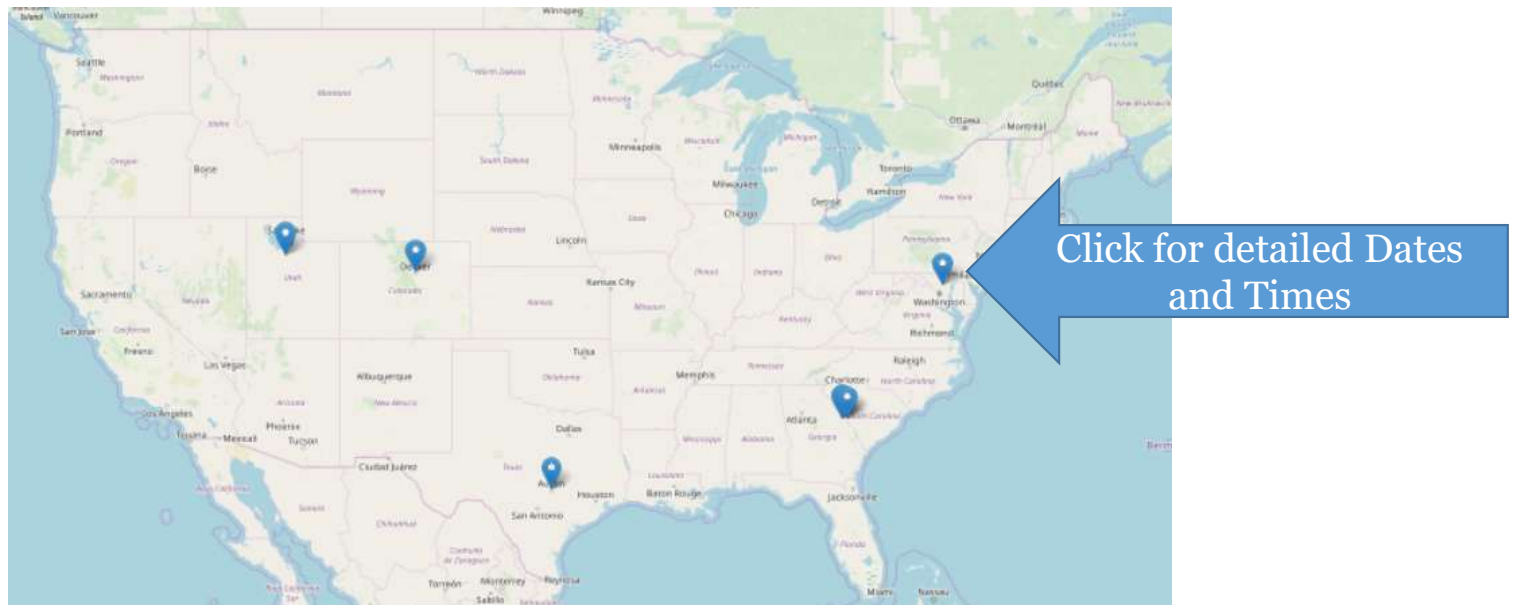
FOR500: Windows Forensic Analysis



GIAC Certified Forensic Examiner (GCFE)

Werejugo

- Acquires wireless information from Registry, WLAN Autoconfig logs, Network Diagnostic events and SRUM



- <https://github.com/markbaggett/werejugo>

One Last Class to Mention

- I'm happy to support these tools!
- I enjoy writing tools and I hope these are truly useful to you.
- But more than anything I want to teach you to write your own tools.
- Your team needs to operate at the "speed of incident"
- Don't let your security teams capability be limited by what tools someone else has or has not written!
- Give a man a fish, and you feed him for a day; show him how to catch fish, and you feed him for a lifetime. - Proverbs 12:10

**SEC573: Automating
Information Security with
Python**



GIAC Python Coder (GPYC)

Visit Mark's Profile Page for His Additional Resources

sans.org/profiles/mark-baggett/



Twitter @MarkBaggett

ADDITIONAL CONTRIBUTIONS BY MARK BAGGETT:

WEBCASTS

[New tools for your threat hunting toolbox](#), August 2020

[The Hackers Apprentice](#), May 2020

[Check out SEC573! More Python3! More Pywars!](#), April 2020

[SANS Introduction to Python Course](#), August 2019

PRESENTATIONS

[TEDxAugusta | Pay no attention to the hacker behind the curtain](#)

[Security Weekly #471 - Mark Baggett, SANS](#)

[KringleCon - Escaping Python Shells](#)

TOOLS

- [eapmd5crack.py](#) - A python implementation of an EAP authentication cracking.
- [Freq Server](#) - A Web server that integrates with SEIM systems and identifies hosts being used for Command and control by identifying domains being used for Command and Control. The tools uses character frequency analysis to identify random hostnames.
- [Domain Stats](#) - A SEIM Integration tool that monitors DNS hostnames used by your network to identify first contact with new domains and contact with new domains that have been established in the last 2 years, effective in identifying malicious actors.
- [API-ify](#) - A Web server that provides an API that allows network defenders to consume the output of any Linux based command and integrate it into their ELK stack, splunk or other SEIM tools.